Distributed Topology Control Algorithm for Multihop Wireless Networks

Steve A. Borbash

Laboratory for Telecommunications Sciences
2800 Powder Mill Road
Adelphi, MD 20783-1197
saborba@lts.ncsc.mil

Esther H. Jennings
Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, CA 91109
esther.jennings@jpl.nasa.gov

Abstract - We present a network initialization algorithm for wireless networks with distributed intelligence. Each node (agent) has only local, incomplete knowledge and it must make local decisions to meet a predefined global objective. Our objective is to use power control to establish a topology based on the relative neighborhood graph which has good overall performance in terms of power usage, low interference, and reliability.

I. Introduction

In a multihop wireless network, a packet may need to be sent over several consecutive wireless links to reach its destination. Multihop networks have the advantage of saving power; as the distance increases, the transmission power required to maintain the same signal-to-noise level increases as a quadratic function of the distance. In addition, multihop networks can overcome obstacles and enhance spatial reuse. The question is, "how should the nodes be connected to achieve good overall performance?" To evaluate performance of a wireless network, some of the suggested metrics are: throughput, delay, power utilization, network connectivity, interference, and reliability. Let us consider each of these metrics from a graph theoretic viewpoint.

Throughput and interference are related; by reducing interference, we can obtain more spatial reuse, and a higher throughput. Interference occurs when a node and its neighbor(s) are transmitting simultaneously. Thus, a topology with a small bound on the node degrees will reduce interference. Two nodes are neighbors if there is an edge connecting them in the topology. Note that, this alone may not be sufficient to avoid interference. We can use topology to limit the number of neighbors a node should communicate to. However, this does not mean the node, when transmitting, will not interfere with other non-neighbor nodes. Thus, topology must be used together with scheduling to avoid interference.

Delay depends on the speed of propagation and the number of hops a packet must travel to reach its destination. The hop-diameter of a network is the maximum number of hops among the shortest paths considering all possible node pairs. By reducing the hop-diameter of a network and routing packets on the shortest paths, we can reduce delays. An issue we do not consider here is network traffic. Given a topology, it may be desirable to avoid routing on shortest paths if this creates congestion or hot-spots in the network. We are mainly concerned with the task of obtaining a good topology for communication. The scheduling and routing problems on such a topology are solved separately.

In networks where the nodes operate on limited battery power, it is important to minimize power consumption to prolong the network's life time. To minimize power, we should exclude long edges and include short edges whenever possible, while optimizing the hop-diameter and maintaining network connectivity/biconnectivity. This led to approaches using the Voronoi diagram and nearest neighbor graphs with directional information[1], [2]. It has also been shown that one can optimize the maximum power used by performing power adjustments while guaranteeing network connectivity and biconnectivity [3].

The connectivity among nodes directly influences the efficiency of information dissemination and routing in that network. Conventionally, the topology of an ad hoc network without power control is defined by the transmission power and the data rate. Assuming a fixed data rate, the fixed transmission power implies a fixed transmission radius. Due to the ad hoc nature of such networks, using a fixed transmission radius might not render a connected network. To increase network robustness against node failures, it is not enough to require that the network topology be connected, but it should be biconnected [3].

In this paper, we design a topology control algorithm with distributed intelligence to construct a topology with the following optimization objectives:

The research described in this publication was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

- minimize node degrees.
- minimize the hop-diameter of the network.
- minimize the maximum transmission radius.
- guarantee connectivity.
- minimize the number of biconnected components.
- maximize the size of the largest biconnected component.

Each node in the network acts as an agent. Each agent has only local (incomplete) knowledge and it must make local decisions to meet certain pre-defined global objective(s). Our goal is to obtain a topology with good graph properties such that it is dense enough to be robust (biconnectivity) and sparse enough to enable spatial reuse (reducing interference, the number of time slots used in TDMA, or the number of channels needed in CDMA).

II. Comparison to Previous work

Recently, several results concerning wireless topology discovery and control have been reported[1]-[4]. A Bluetooth Topology Construction Protocol (BTCP) was proposed in[5]. BTCP uses multiple channels and frequency hopping. The problem is to determine which nodes should share a common channel such that the resultant graph induced by all of the nodes is connected. BTCP requires synchronized frequency hopping patterns for the nodes to discover each other. The nodes sharing a common channel form a piconet. Each piconet has a master and a limited number of slaves. The masters of piconets are elected by a distributed leader election algorithm. The piconets are connected by bridges to form a connected wireless network. The optimization objective of BTCP is to minimize the time needed for network initialization, that is, to establish the links. Power issues and graph properties are not addressed.

A family of probabilistic protocols, called birthday protocols, was presented in [6]. In these protocols, two wireless nodes independently and randomly select k slots out of n time slots. The first node transmits during the k slots it selected, and the second node listens during its k slots. A node is idle during the other (not selected) $n - k$ time slots. It was shown that even when k/n is a small value, the probability of the second node hearing the first node is almost one, and yet the nodes are idle most of the time. This network topology discovery method is very energy efficient. Although the birthday protocols do not guarantee the discovery of all nodes within the transmission range, the probability of a node being discovered is high; that is, there are very few undiscovered nodes. The birthday protocols do not use power adjustments to control the network topology.

Hu [1] suggested a distributed topology algorithm for packet radio networks, based on the Voronoi diagram and the Delaunay triangulation. A Delaunay triangulation is one that maximizes the minimum angle. Intuitively, this makes the triangles more equilateral, and hence minimizes the node degree. Two parameters are used to control the topology: Δ and R, where Δ controls the node degree and R controls the transmission radius. Given the Delaunay triangulation, edges longer than R are removed. If there are nodes whose degree exceeds Δ, the longest edges incident to these nodes are removed to bring the node degrees down to Δ. When a node abandons an edge, it will notify the neighbor which is incident to that edge. To achieve good connectivity, edges not in the graph are added (from the shortest to the longest, not exceeding length R) in such a way that the node degree of $\leq \Delta$ is preserved. Matching is used when adding an edge. The distributed implementation of this algorithm relies on each node computing a part of the Delaunay triangulation containing all the nodes reachable within a radius R. The topology produced has better performance in terms of throughput and reliability, compared to topologies using a fixed R only or a fixed Δ only (connecting to the nearest Δ nodes). The Delaunay triangulation is a proximity graph*. In [7], [4], geometric spanners were used as power efficient routing structures. These geometric spanners are closely related to proximity graphs. Encouraged by these results, in this paper, we present a distributed algorithm for topology control based on a specific proximity graph, the relative neighborhood graph. We will motivate this choice later.

In [3], it was proposed to assign different transmit powers to different nodes to meet a global topological property such as connectivity and biconnectivity. The objective is to minimize the transmit power. A centralized algorithm constructs the topology in a manner similar to the building of the minimum spanning tree, that is, by adding one edge at a time such that the added edge connects different components. Since each edge represents a transmission radius, there are side-effect edges which are added as well. Then, a per node minimization is made to remove extra edges by reducing power, while maintaining connectivity/biconnectivity. Two distributed heuristics for topology control were proposed. One heuristic uses locally available neighbor information collected by a routing protocol to keep the node degrees bounded. The other heuristic uses locally available neighbor information and global topology information such as those provided by link-state protocols.

Note that, both of the methods in [1] and [3] use proximity graphs: minimum spanning tree [3], and Delaunay triangulation [1]. What is interesting here is that the proximity graph we choose fits in-between the minimum spanning tree and the Delaunay triangulation. It has been shown that, for a given node set V in an Euclidean plane, $MST(V) \subseteq RNG(V) \subseteq DT(V)$, where MST, RNG and DT denote the minimum spanning tree, the relative neighborhood graph and the Delau-

*Proximity graphs is a family of graphs where the edges of the graphs correspond to different notions of closeness (or proximity) between the nodes' geometric placements.

nay triangulation respectively [8]. We choose the RNG because it represents the internal structure, relative closeness, of the node set. RNG is also more flexible than MST and DT. In MST and DT, the edges between nodes are determined only by the absolute distances. In addition to this, the RNG takes into account the relative distance of each pair of nodes to the remaining nodes. While MST is a tree, and DT is a collection of triangle faces (in the non-degenerate cases), RNG ranges from a tree to a full triangulation depending on the relative distances of nodes. In this sense, RNG gives a good representation of how the nodes relate to each other when seen as a whole.

III. Network Topologies

In a previous study [9], we compared the graph properties of the minimum spanning tree (MST), the relative neighborhood graph (RNG), and the minimum radius graph (minR). Given a node set V in an Euclidean plane, the MST is a tree containing all the nodes of V, such that the total edge length of the tree is minimized. Assuming all the nodes must use the same transmission radius, the minR graph is obtained by finding the smallest radius such that connectivity is achieved. When the smallest radius r is found, the topology is defined by connecting all the nodes that are within a distance r from each other. Let $l_i, l_j \in IR^2$ be the locations of nodes v_i and v_j respectively, where $v_i \neq v_j$. The RNG is the graph $G = (V, E)$, where $(v_i, v_j) \in E$ if and only if there is no node $v_z \in V$ such that $\| l_i - l_z \| < \| l_i - l_j \|$ and $\| l_j - l_z \| < \| l_i - l_j \|$, or equivalently, the edge between nodes v_i and v_j is valid if there does not exist any node closer to both v_i and v_j. Given the same 50 random nodes uniformly distributed in a 600×600 square plane, Figures 1, 2, 3 show the MST, the RNG and minR respectively. As we can see from these Figures, minR is dense with high node degrees, while MST and RNG are sparse, having low node degrees. However, the minR has a low hop-diameter of 4 while MST and RNG have hop-diameters of 20 and 15, respectively. In this example, all the nodes in minR belong to the same biconnected component, making the minR fault-tolerant. In the RNG, 45 of the 50 nodes are in the same biconnected component, so the majority of the network is fault-tolerant. The MST is 1-connected, so it is not meaningful to consider it for biconnectivity; it is not fault-tolerant.

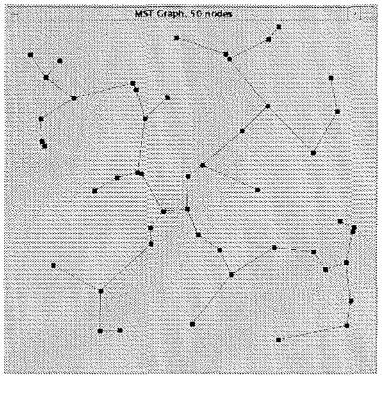

Fig. 1. Minimum Spanning Tree with 50 nodes

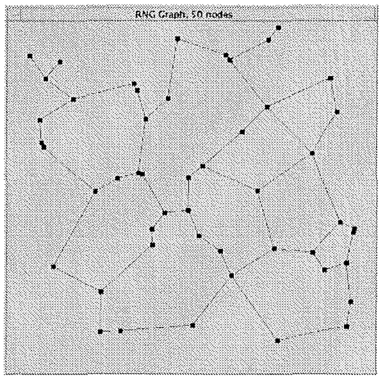

Fig. 2. Relative Neighborhood Graph with 50 nodes

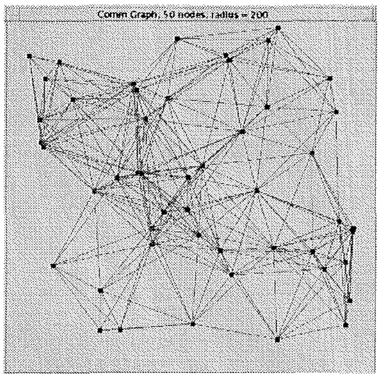

Fig. 3. Minimum (fixed) Radius Graph with 50 nodes

In a previous study [9], we simulated random placements of n nodes where $5 \leq n \leq 800$. For each n value, we generate 1000 different node placements to compute MST, RNG and minR. From the simulation results, we found that:

- **transmission radius**: MST has the smallest average transmission radius while minR has the largest average transmission radius. Although the average transmission radius of RNG is between those of the MST and the minR, it approaches the value of the MST's transmission radius as the number of nodes increases.

- **hop-diameter**: on the average, minR has the smallest hop-diameter and MST the largest, where RNG is in-between the two. However, it is worth noting that the value of RNG's hop-diameter is closer to minR's hop-diameter than to MST's.
- **node degree**: both of MST and RNG have node degrees bounded by a small constant (approx. 6), where the node degree of minR increases linearly as the number of nodes increases.
- **biconnectivity**: in our previous simulation runs, minR is always biconnected. The RNG has 86% of the nodes in the same biconnected component, when $n = 100$ and over 90% of the nodes in the same biconnected component, when $n = 200$, where n is the number of nodes.

Based on the above observation, we see that MST has good performance in terms of transmission radius and node degree. On the other hand, minR scores well in terms of hop-diameter and biconnectivity. Interestingly, where MST performs well, RNG's performance is close to MST's; and where minR performs well, RNG's performance is close to minR's. From this, we consider RNG as a desirable topology for wireless networks in achieving our topology objectives listed in Section I.

IV. Wireless Network Model

Given a set V of n nodes in a Euclidean plane, we assume the following concerning the network model.

- Each node has limited battery power.
- Each node uses an omni-directional antenna for communication.
- As in [2], we assume each node can sense the direction of incoming signals from neighboring nodes.
- Similar to [2], [3], we assume that transmit power p can be set to any positive level such that $0 \leq p \leq MAX$, where MAX is the maximum power level. Any node within the transmission radius of a node v can hear v.
- A node can send a broadcast and every node that hears the broadcast can send an acknowledgment. Similar to [2], we assume the existence of an underlying MAC layer that resolves interference.
- Interference to any nodes outside of the transmission radius is considered negligible.

V. Algorithm

A simplest brute-force algorithm for computing the RNG is as follows. First, compute $\| l_i - l_j \|$ for all $\binom{n}{2}$ possible node pairs v_i, v_j, where $v_i \neq v_j$. Then, for each of the possible edges (v_i, v_j) and each node v_k, $v_k \neq v_i$ and $v_k \neq v_j$, if $\| l_i - l_k \| < \| l_i - l_j \|$ and $\| l_j - l_k \| < \| l_i - l_j \|$ then exclude edge (v_i, v_j) from the RNG. This brute-force algorithm is impractical because it requires $\Theta(n^3)$ computation steps. It also requires global information at every node. Knowing that RNG is a subgraph of the DT, and that DT can be computed in $O(n \log n)$ steps, we can first compute DT and then check each edge of the DT with the other nodes to determine if it is an edge of RNG. Since DT is a planar graph, the number of edges in DT is linear with respect to n, thus, this method requires $O(n^2)$ steps to compute RNG. Supowit[10] designed the first $O(n \log n)$ sequential time algorithm for RNG. The method scans the nodes from six different directions, each direction is separated by an angle of $\frac{\pi}{3}$.

We present a novel distributed algorithm using local information and directional information of incoming signals. The algorithm is executed at each node, and it can run asynchronously. There are two main steps of the algorithm. Initially, the transmission power p is set to zero and the entire 2π angle around the node spans the not-yet-covered region extending from the node. Let Θ denote the set of angles which define cones that jointly span the covered region(s); initially, Θ contains a single element θ_{zero} which is an angle of value 0.

Algorithm Dist_RNG

Input : a set of V of n nodes.

Output : RNG(V), where the edge lengths are bounded by the maximum transmission power. We assume all the nodes start the algorithm simultaneously. This condition can be relaxed as we will discuss later.

Step 1 : A node v_i grows its transmission power until a nearest neighbor v_j is found in the not-yet-covered region. Add the edge (v_i, v_j) to RNG. If there are several nearest neighbors reachable at the same transmission power level (on the circumference of the circle centered at v_i with power radius p), then add the edges from v_i to each of the reachable nodes to the RNG.

Step 2 : use the newly found nearest neighbor v_j to compute the angle θ_j, where θ_j defines a cone that spans the area covered by v_j. Update Θ by merging θ_j with Θ. Note that the resulting Θ may span non-adjacent, disjoint regions.

Repeat Steps 1 and 2, until Θ contains angles whose cones jointly span the entire 2π region around v_i, or when maximum power is reached.

Lemma 1: At any time during the execution of Algorithm Dist_RNG, the nearest neighbor v_j found by v_i in a not-yet-covered region defines an angle θ_j such that at most one of $\theta_j \cap \theta_x > \frac{\pi}{3}$, for all $\theta_x \in \Theta$.

Proof: When the entire 2π region around v_i is not-yet-covered, $\Theta = \{\theta_{zero}\}$, so $\theta_j \cap \theta_{zero} < \frac{\pi}{3}$.

Now, consider the case where Θ contains one non-zero an-

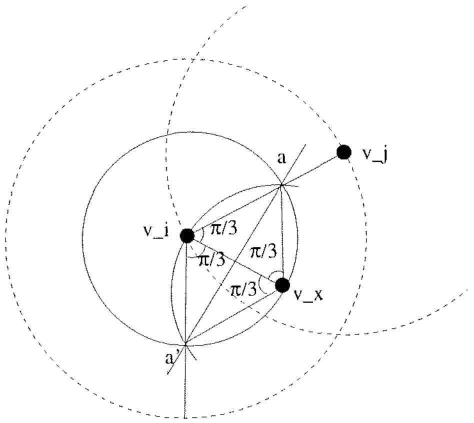

Fig. 4. intersection of spanning angles

gle θ_x, where α can result from the angle spanned by one neighbor, or the merged angles spanned by several neighbors. Suppose θ_x is spanned by one neighbor node v_x, as shown in Figure 4. Draw a circle with radius $\| l_i - l_x \|$ centered at v_x; this circle will intersect the circle with the same radius centered at v_i. The intersection is called a *lune*. Draw a line that bisects the edge (v_i, v_x). This line will intersect the lune at the two apexes a and a' as shown. Note that the triangles a, v_i, v_x and a', v_i, v_x are equilateral triangles. Thus, the spanning angle $\theta_x = \angle a' v_i a = \frac{2\pi}{3}$. Furthermore, the cone spanned by θ_x, extending from v_i can be considered as a covered region, because v_j eliminates any node v_k in the region to be a RNG neighbor of v_i. Note that the lune of (v_i, v_x) is empty, then any node v_k in the covered region will satisfy both of the following conditions: (a) $\| l_i - l_x \| < \| l_i - l_k \|$, and (b) $\| l_x - l_k \| < \| l_i - l_k \|$. Thus, v_x eliminates (v_i, v_k) to be in the RNG. Since the next nearest neighbor node v_j also defines a $\frac{2\pi}{3}$ spanning angle θ_j, where (v_i, v_j) bisects θ_j, and v_j cannot be in the region covered by v_x, the largest overlap of θ_x and θ_j must be bounded by $\frac{\pi}{3}$. Similarly, if Θ contains one angle which is the result of several nearest neighbor nodes found previously, then there must be a node v_x which defines an angle which has the largest overlap with v_j. The same argument applies.

If Θ contains more than one angle, then the angles cover non-adjacent disjoint regions. Since each nearest neighbor node defines a $\frac{2\pi}{3}$ angle, we can have at most two such angles spanning non-adjacent, disjoint regions. Let θ_x, θ_y be the two angles in Θ. For a 2π region, we can have four adjacent regions r_1, r_2, r_3, r_4. Let r_1, r_3 be the regions covered by θ_x, θ_y respectively. Let θ_a, θ_b be the angles covering r_2, r_4 respectively. Then $\theta_a + \theta_b = \frac{2\pi}{3}$. If $\theta_a = \theta_b = \frac{\pi}{3}$, then a nearest neighbor v_j in any uncovered region will define an angle θ_j such that $\theta_j \cap \theta_x \leq \frac{\pi}{3}$, for any angle in Θ. However, if $\theta_a < \theta_b$, and v_j defines an angle θ_j that contains θ_a, then there can be at most one $\theta_j \cap \theta_x > \frac{\pi}{3}$. If $\theta_a < \theta_b$, and v_j defines an angle θ_j that intersects with θ_b, then $\theta_j \cap \theta_x \leq \frac{\pi}{3}$. This will take away at least $\frac{\pi}{3}$ from θ_b such that the remaining not-yet-covered region is spanned by $\theta_{b'} < \frac{\pi}{3}$. The algorithm will continue to increase the transmission power. When the next nearest neighbor $v_{j'}$ is found, there is exactly one $\theta_{j'} \cap \theta_x > \frac{\pi}{3}$, for all $\theta_x \in \Theta$.

Corollary 2: At any time during the execution of Algorithm Dist_RNG, the nearest neighbor in a not-yet-covered region is an edge of the RNG.

Proof: This can be derived from Lemma 1, because the node in the not-yet-covered region has not been eliminated by any other node in the covered region, and because the transmission power of the node looking for a neighbor is monotonically increasing.

Theorem 3: Algorithm Dist_RNG computes a relative neighborhood graph correctly in six rounds.

Proof: The algorithm is correct because it always connects a node to its next nearest neighbor(s) which is (are) not yet eliminated. Whenever one or more neighbors are found at the same transmission radius, this constitutes one round of the algorithm. At a node, the entire 2π angle is not-yet-covered initially. Suppose one neighbor is found at each round producing one angle in Θ, this takes away at least $\frac{\pi}{3}$ from the not-yet-covered angle (because of Lemma 1). In this scenario, the algorithm finishes in six rounds. Suppose one neighbor is found at each round but producing two angles in Θ in two rounds. Then, there are two cases to consider.

Case 1: the two remaining cones are spanned by angles of different sizes. Then, the algorithm can take another three rounds to complete, one round to cover the smaller of the remaining not-yet-covered angles, and two rounds to cover the larger of the remaining not-yet-covered angles.

Case 2: the two remaining cones are spanned by angles of the same size. Then each angle must be of size $\frac{\pi}{3}$, and each of these can be covered in one round.

Suppose more than one neighbor is found in a round, this does not cause any extra rounds. Thus, the number of rounds is bounded by six.

Note that, we assume the maximum power is large enough so that the resulting RNG is connected. If the area is large, the number of nodes is small, and the maximum power is small, then we cannot guarantee a connected graph. From our previous simulation results, the expected edge length of the edges in a RNG is $O(\sqrt{\frac{A}{n}})$, where A and n are the area size and the number of nodes respectively. Setting the power to be slightly above that would often produce a connected graph.

We now consider the relaxed initial condition and dynamic adaptation of the algorithm. We have assumed a strong constraint that all the nodes start the algorithm simultaneously.

This is so that we can analyze the number of rounds needed easily. However, this algorithm will also work when at least one node starts the algorithm. All other nodes can be in the listen mode initially. As soon as a node hears a message, it will start its local algorithm to search for neighbors. In this scenario, we must add the cost to start the local algorithm at all the nodes. Since the algorithm is local and does not require global information, in mobile ad hoc networks, the recomputation is carried out only at the nodes affected by the moved nodes.

VI. Conclusion and Application

We have motivated that the relative neighborhood graph can be a good candidate for topology control due to its good graph properties in terms of throughput, interference, delay, power and connectivity. We also designed a novel distributed algorithm to compute the relative neighborhood graph using only local and directional information. We intend to implement simulations of the algorithm and compare it to other topologies based on proximity graphs such as the minimum spanning tree and the Delaunay triangulation.

A possible application of using power control to obtain a relative neighborhood graph as the underlying communication graph for wireless networks is the following. In [11], Quirk et al. proposed to use cooperative modulation techniques for long haul relay in space exploration missions where sensor networks are used on the surface of the planet being explored. By sharing the information to be transmitted to the satellite among the sensor nodes, they can cooperate to reduce the total energy needed to transmit the data from the surface of a planet to orbit, thus extending the lifetime of the energy-restricted sensor nodes. They presented and showed that the node-selection on orthogonal channels (NSOC) scheme offers significant energy savings over the non-cooperative communication method. For local communication, they have only considered line and grid topologies. Our result is applicable to the NSOC method by considering arbitrary topologies imposed by random placement of the nodes. For example, knowing the size of the bounding area A which contains all the sensor nodes, and the number of sensor nodes n, we can obtain a topology with good performance in terms of interference and hop-diameter. This can be used as the communication topology for local communication among the sensor nodes prior to the cooperative long-haul communication to the satellite.

For future work, we propose to compare different proximity graphs to determine their suitability as wireless network topologies considering power consumption, scheduling and routing. The comparisons can be made by theoretical analysis and simulations using different distributions. We also propose that new metrics be defined for measuring the distance between two wireless nodes. For example, the power needed to support a link (a, b) may be $\| ab \|^2$ or $\| ab \|^4$ depending on the model used for power dissipation. In the presence of obstacles, the Euclidean distance (or a function of this) may not be an appropriate representation of the power needed to support a link between two nodes. Thus a new graph model or new distance metric is needed.

References

[1] Limin Hu, "Topology control for multihop packet radio networks," *IEEE Transactions on Communications*, vol. 41, no. 10, pp. 1474–1481, October 1993.

[2] Roger Wattenhofer, Li Li, Paramvir Bahl, and Yi-Min Wang, "Distributed topology control for power efficient operation in multihop wireless ad hoc networks," in *Proc. IEEE INFOCOM 2001 Conference*, April 22-26 2001, Anchorage, Alaska.

[3] Ram Ramanathan and Regina Rosales-Hain, "Topology control of multihop wireless networks using transmit power adjustment," in *Proc. IEEE INFOCOM 2000 Conference*, March 26-30 2000, Tel Aviv, Israel.

[4] Xiang-Yang Li, Peng-Jun Wan, and Yu Wang, "Power efficient and sparse spanner for wireless ad hoc networks," in *Proc. of the 10th IEEE International Conference on Computer Communications and Networks*, October 15-17 2001, Scottsdale, Arizona.

[5] Theodoros Salonidis, Pravin Bhagwat, Leandros Tassiulas, and Richard LaMaire, "Distributed topology construction of Bluetooth personal area networks," in *Proc. IEEE INFOCOM 2001 Conference*, April 22-26 2001, Anchorage, Alaska.

[6] Michael J. McGlynn and Steve A. Borbash, "Birthday protocols for low energy deployment and flexible neighbor discovery in ad hoc wireless networks," in *Proc. ACM Symposium on Mobile Ad Hoc Networking and Computing (Mobihoc) 2001 Conference*, October 4-5 2001, Long Beach, California.

[7] Jie Gao, Leonidas J. Guibas, John Hershberger, Li Zhang, and An Zhu, "Geometric spanners for routing in mobile networks," in *Proc. of the 2nd ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '01)*, October 4-5 2001, pp. 45–55, Long Beach, California.

[8] G. T. Toussaint, "The relative neighborhood graph of a finite planar set," *Pattern Recognition*, vol. 12, pp. 261–268, 1980.

[9] Esther Jennings and Clayton M. Okino, "Topology for efficient information dissemination in ad hoc networking," August 2001, submitted.

[10] Kenneth J. Supowit, "The relative neighborhood graph with an application to minimum spanning trees," *Journal of the ACM*, vol. 30, no. 3, pp. 428–448, July 1983.

[11] Kevin Quirk, Meera Srinivasan, and Jonathan R. Agre, "Cooperative modulation techniques for long haul relay in sensor networks," in *Proc. of GLOBECOMM 2001*, Nov 25-29 2001, San Antonio, Texas.